An Implicit LU-SGS Scheme for the Spectral Volume Method on Unstructured Tetrahedral Grids

Takanori Haga[1,*], Keisuke Sawada[1] and Z. J. Wang[2]

[1] *Department of Aerospace Engineering, Tohoku University, Sendai 980-8579, Japan.*
[2] *Department of Aerospace Engineering, Iowa State University, Ames, IA 50011, USA.*

Abstract. An efficient implicit lower-upper symmetric Gauss-Seidel (LU-SGS) solution approach has been applied to a high order spectral volume (SV) method for unstructured tetrahedral grids. The LU-SGS solver is preconditioned by the block element matrix, and the system of equations is then solved with a LU decomposition. The compact feature of SV reconstruction facilitates the efficient solution algorithm even for high order discretizations. The developed implicit solver has shown more than an order of magnitude of speed-up relative to the Runge-Kutta explicit scheme for typical inviscid and viscous problems. A convergence to a high order solution for high Reynolds number transonic flow over a 3D wing with a one equation turbulence model is also indicated.

AMS subject classifications: 65M70, 76M12, 76M22

Key words: High-order, unstructured grid, spectral volume, implicit method.

1 Introduction

Unstructured grid methods have been widely used in aerodynamic design processes because they can offer flexible grid generation for 3D complex configurations. Aerodynamic coefficients at cruising condition can be predicted reasonably well with steady Reynolds Averaged Navier-Stokes Simulation (RANS) using conventional finite volume solvers of second order accuracy in space. However, some problems become prohibitively expensive to reach sufficient prediction accuracy with increasing grid points. Examples include vortex dominated flows such as flow over high-lift configurations, aero-acoustic noise predictions and LES/DNS for high Reynolds number flowfields. In these flows, in order to resolve important flow features of unsteady vortices, high-order methods are required.

*Corresponding author. *Email addresses:* `haga@cfd.mech.tohoku.ac.jp` (T. Haga), `sawada@cfd.mech.tohoku.ac.jp` (K. Sawada), `zjw@iastate.edu` (Z. J. Wang)

In a conventional finite volume framework, arbitrarily high-order schemes can be obtained theoretically by extending stencils for high-order polynomial data reconstructions, but a crucial difficulty exists in finding valid (non-singular) stencils for unstructured grids. Recently, new high-order unstructured grid methods such as the discontinuous Galerkin (DG) method [5,6], the spectral volume (SV) method [9,15,23,27–29,31] and the spectral difference (SD) method [14] have attracted attention. These methods introduce degrees of freedom within each element for a high-order approximation to the solution and are expected to achieve the formal order of accuracy even on an unstructured grid.

The SV method belongs in a class of Godunov-type finite volume methods, and yet the degrees-of-freedom are introduced by partitioning a grid cell. If a proper partition is found, a valid reconstruction stencil comprised of partitioned sub-cells can be predetermined. In our previous study [8], the SV method was developed for the Euler and the Navier-Stokes equations in 3D and we demonstrated its applicability to large-scale parallel computations. However, the developed solution technique was CPU demanding and inefficient for steady state problems. The excessively high CPU cost was attributed mainly to the use of an explicit Runge-Kutta time integration scheme. The Runge-Kutta scheme is easy to implement and requires only a small amount of memory, but suffers from slow convergence, especially for viscous problems in which grid points are clustered in the boundary layer.

Several efficient time integration/iterative solution approaches for the DG method have been developed, for example, preconditioned GMRES approaches [2,17], a p multigrid approach using a line implicit smoother [7] and a low storage p multigrid approach [16]. Recently, an efficient LU-SGS scheme [4] has been developed for the SD method [24]. Implicit schemes for the SV method on tetrahedral grids, however, are yet to be developed.

In this study, the LU-SGS approach is applied to the SV method on unstructured tetrahedral grids. The LU-SGS solver is preconditioned by the block SV-element matrix, and the system of equations is then solved with an exact LU decomposition linear solver. The developed method is tested first for a linear advection problem, and then for typical steady problems by solving the Euler and Navier-Stokes equations. A significant reduction in computational time comparing with the multi-stage Runge-Kutta scheme is demonstrated. Finally, we show the converged solution of viscous transonic flow over a 3D wing, where the effect of turbulence is accounted for by solving a one-equation turbulence model using a SV scheme, to indicate the applicability of the present implicit SV scheme for practical engineering problems.

2 Numerical methods

2.1 Spectral volume discretization

The unsteady, 3D, compressible Navier-Stokes equations in conservative form can be expressed as

$$\frac{\partial Q}{\partial t} + \nabla \cdot F(Q) - \nabla \cdot F_v(Q, \nabla Q) = 0, \tag{2.1}$$

where $Q = (\rho, \rho u, \rho v, \rho w, e)^T$ is the conservative state vector, $F(Q) = (F^x, F^y, F^z)$ is the inviscid flux, and $F_v(Q, \nabla Q) = (F_v^x, F_v^y, F_v^z)$ the viscous flux.

As with other generalized Godunov methods, the spectral volume (SV) method mainly consists of two steps: local data reconstruction, and evaluation of inviscid and viscous fluxes. In the SV method, the computational domain is divided into non-overlapping tetrahedral cells in 3D called spectral volumes (SVs), and each SV is further sub-divided into "structured" sub-cells, called control volumes (CVs). The number of CVs in one particular SV depends on the degree of the polynomial of the reconstructed dependent variables. For a complete polynomial basis in 3D, a reconstruction for degree of precision p requires at least N CVs in a SV, which is $N(p) = (p+1)(p+2)(p+3)/6$. The reconstructed piecewise polynomial $\tilde{q}_i$ in the i-th SV can be expressed using cell averaged data $\bar{q}_{i,j}$, $(j = 1, \cdots, N(p))$ on these CVs as,

$$\tilde{q}_i(\vec{r}, t) = \sum_{j=1}^{N(p)} L_j(\vec{r}) \bar{q}_{i,j}(t), \tag{2.2}$$

where L is a set of polynomials that satisfies the following algebraic equations,

$$\frac{1}{V_j} \int_{CV_j} L_l(\vec{r}) dV = \delta_{j,l} \quad (l = 1, \cdots, N(p)). \tag{2.3}$$

The set of polynomials L is referred to as shape functions, and can be determined for non-singular partitions of SV. Although every SV shape could be different in the physical space, an identical set of shape functions is available in a standard space using linear transformations. Therefore, the subscript i for L in Eq. (2.3) is suppressed. It is known that the stability and convergence properties of the SV method heavily depend on the partition. The Lebesgue constant of the reconstruction matrix is one of the measures to assess the quality of the partition. Recently, some improved partitions were found using linear analysis by Van den Abeele and Lacor [25], but an optimum partition for 3D tetrahedral SV elements is still unknown. In this study, we use the linear ($p1$) and quadratic ($p2$) partitions given by Chen [3], that was chosen to have a small Lebesgue constant.

The NS equations are discretized by integrating them over the j-th CV in the i-th SV named $CV_{i,j}$ as

$$\frac{d\bar{Q}_{i,j}}{dt} + \frac{1}{V_{i,j}} \int_{\partial CV_{i,j}} F(Q) \cdot \vec{n} dS - \frac{1}{V_{i,j}} \int_{\partial CV_{i,j}} F_v(Q, \nabla Q) \cdot \vec{n} dS = 0. \tag{2.4}$$

where $\bar{Q}_{i,j}$ represents the cell averaged state vectors over $CV_{i,j}$, $V_{i,j}$ is the volume of $CV_{i,j}$, and $\vec{n}$ is the outward unit normal vector of the boundary $\partial CV_{i,j}$. The cell averages of

conservative variables on CVs are evolved in time as solution unknowns. Because Q is assumed to be a piecewise polynomial in each SV, the value is discontinuous at SV boundaries. For the inviscid flux, the approximate Riemann solver is applied. On the other hand, the flux at the CV boundaries inside a SV can be obtained analytically because the reconstructed distributions are continuous there.

Like the DG method, the discretization of the viscous term is not straightforward with the SV method because dependent variables are discontinuous at SV boundaries. The so-called naïve formulation, in which the gradient value at the interface is determined as an average of gradient values in adjacent SV cells sharing the interface, is known to converge to a wrong solution [22]. Recently, the second approach of Bassi and Rebay (BR2) [2] for DG method was adopted in the SV method [11] and was shown to have a wider stability limit for the time step than the local DG approach and the penalty approach for the 2D Laplace equation. In this study, we employ the BR2 formulation to discretize the viscous term.

The local lifting operator for a SV face S is defined in weak form as

$$\int_{SV^-} \vec{r}^- dV = \int_S \vec{n}^- \frac{\tilde{Q}^+ - \tilde{Q}^-}{2} dS, \tag{2.5}$$

$$\int_{SV^+} \vec{r}^+ dV = \int_S \vec{n}^+ \frac{\tilde{Q}^- - \tilde{Q}^+}{2} dS, \tag{2.6}$$

where the subscripts $-$ and $+$ denote the "interior" and the "exterior" of the face, respectively, and $\vec{n}^-$ is the unit normal vector directing from "interior" to "exterior". Let us assume that $\vec{r}$ can be written as a p-th order polynomial in the SV cell. Using the property of the shape functions in the LHS of Eq. (2.5), we obtain

$$\bar{\vec{r}}^-_{,j} V^-_{,j} = \int_{S \cap \partial CV_j} \vec{n}^- \frac{\tilde{Q}^+ - \tilde{Q}^-}{2} dS, \tag{2.7}$$

where $\bar{\vec{r}}_j$ is the CV averaged value. Note that the surface integral in the RHS of Eq. (2.7) is computed only on the CV boundary on the SV interface in order to assure the local property. Finally, the locally "averaged" gradients at the SV face are computed as

$$\tilde{G} = \frac{\nabla Q^- + \nabla Q^+}{2} + \eta \frac{\vec{r}^- + \vec{r}^+}{2}, \tag{2.8}$$

where η is a positive constant required to ensure the numerical stability for elliptic problems. η must be chosen to be at least as large as the total number of faces of an element, i.e., in the case of tetrahedral meshes, $\eta \geq 4$. In this study, η is set to be 4.

The flux integration on each face is evaluated using the Gauss quadrature formula [13] in order to achieve the designed order of accuracy. For the second order SV scheme using $p1$ polynomials, fluxes only at the centroid of each CV face are required as in the case for second order finite volume schemes. For the third order SV scheme using $p2$ polynomials, polygonal faces are subdivided into a union of triangles and fluxes at the

Gauss quadrature points which are the midpoints of triangle edge are calculated. The formal spatial accuracy of the resulting SV scheme is $p+1$ using p-th order polynomial reconstruction. An apparent drawback of the Gauss quadrature approach is the high computational cost because the number of quadrature points increase rapidly when p is greater than 1. In fact, the number of quadrature points for a SV is 16 for a $p1$ polynomial and 160 for a $p2$ polynomial. Note that a substantial improvement in computational efficiency can be obtained by following the quadrature free approach [9]. Further details of the SV method can be found in the references [15, 23, 28, 31].

2.2 Implicit non-linear LU-SGS method

We define the spatial residual operator for a control volume $CV_{i,j}$ as

$$R_{i,j}(\bar{Q}) = -\frac{1}{V_{i,j}} \int_{\partial \mathrm{CV}_{i,j}} (F - F_v) \cdot \vec{n} dS. \tag{2.9}$$

Applying the backward Euler differencing for the time derivative in Eq. (2.4) gives a nonlinear system of

$$\frac{\bar{Q}_c^{n+1} - \bar{Q}_c^n}{\Delta t} = R_c(\bar{Q}^{n+1}), \tag{2.10}$$

where c represents a current SV cell. Note that the size of vectors $\bar{Q}_c$ and R_c is (the number of dependent variables) $\times$ (the number of CVs in a SV). Because R_c depends not only on $\bar{Q}_c$ but also on $\bar{Q}_{nb}$ in neighboring grid cells through the numerical flux, a linearization of the residual becomes

$$R_c(\bar{Q}^{n+1}) \approx R_c(\bar{Q}^n) + \frac{\partial R_c}{\partial \bar{Q}_c} \Delta \bar{Q}_c + \sum_{nb \neq c} \frac{\partial R_c}{\partial \bar{Q}_{nb}} \Delta \bar{Q}_{nb}. \tag{2.11}$$

Substituting Eq. (2.11) into Eq. (2.10), we obtain

$$\left(\frac{I}{\Delta t} - \frac{\partial R_c}{\partial \bar{Q}_c} \right) \Delta \bar{Q}_c - \sum_{nb \neq c} \frac{\partial R_c}{\partial \bar{Q}_{nb}} \Delta \bar{Q}_{nb} = R_c(\bar{Q}^n). \tag{2.12}$$

Since storing all the implicit Jacobian matrices in the algebraic system requires too much memory, we employ a LU-SGS scheme originally developed by Jameson and Yoon [10]. In order to avoid storing the Jacobian matrices for the "nb" cells, we further introduce inner iteration. The Eq. (2.10) can be rewritten as

$$\frac{\bar{Q}_c^{k+1} - \bar{Q}_c^n}{\Delta t} = R_c(\bar{Q}^{k+1}), \tag{2.13}$$

where k indicates the inner sweep number of the symmetric forward and backward sweeps. The Taylor expansion in terms of the current cell yields

$$R_c(\bar{Q}^{k+1}) = R_c(\bar{Q}_c^{k+1}, \bar{Q}_{nb}^{k+1}) \approx R_c(\bar{Q}_c^k, \bar{Q}_{nb}^{k+1}) + \frac{\partial R_c}{\partial \bar{Q}_c} (\Delta \bar{Q}_c^{k+1} - \Delta \bar{Q}_c^k). \tag{2.14}$$

Substitution of Eq. (2.14) into Eq. (2.13) yields,

$$\left(\frac{I}{\Delta t}-\frac{\partial R_c}{\partial \bar{Q}_c}\right)\left(\Delta \bar{Q}_c^{(k+1)}-\Delta \bar{Q}_c^{(k)}\right)=R_c(\bar{Q}_c^k,\bar{Q}_{nb}^{k+1})-\frac{\Delta \bar{Q}_c^k}{\Delta t}\approx R_c(\bar{Q}_c^k,\bar{Q}_{nb}^*)-\frac{\Delta \bar{Q}_c^k}{\Delta t}, \qquad (2.15)$$

in which we replace $\bar{Q}_{nb}^{k+1}$ by the most recent solution $\bar{Q}_{nb}^*$ in neighboring cells. Note that the Jacobian matrix $\frac{\partial R_c}{\partial \bar{Q}_c}$ is approximately frozen to the value at n-th time step in the above derivation. The solution is updated with multiple symmetric forward and backward sweeps in the domain. For better convergence rate, we employ a reordering technique [20] to determine hyperplanes. For steady state problems it is not necessary to solve Eq. (2.15) to machine zero. If the RHS of Eq. (2.15) in a sweep decreases one order of magnitude with respect to the initial one at each time step, or the number of sweeps reaches to a prescribed maximum number of sweeps $k_{\max}$, we terminate the inner iteration and move to the next time step.

In this study, the Jacobian matrices are numerically obtained using the following numerical differencing

$$\frac{\partial R_c}{\partial \bar{Q}_c}\approx\frac{R_c(\bar{Q}_c+\delta Q,\bar{Q}_{nb})-R_c(\bar{Q}_c,\bar{Q}_{nb})}{\delta Q}, \qquad (2.16)$$

where δQ is a small value scaled by a norm of the difference between the computed conservative variables and the free-stream values as,

$$\delta Q=\|Q-Q_\infty\|\times\delta, \qquad (2.17)$$

where δ is set to be 10^{-4} unless specified explicitly in this study. At the beginning of computations with the impulsive start using the freestream condition, each component of δQ is replaced by 10^{-12} to avoid dividing by zero. In several numerical tests, the convergence property is not sensitive to the constant factor δ in the range of $10^{-2}-10^{-8}$. If the difference between the local state and the freestream state is not small, Eq. (2.16) can lead to yield poorly approximated derivatives. It is found that this is really the case for the working variable of the one-equation turbulence model. For this case, we simply limit the magnitude of δQ (not δ) no greater than 10^{-4}. Finding an optimum choice of δQ needs to be thoroughly examined, but it is beyond the scope of this study.

The resulting element matrices are no longer diagonal, and need to be preconditioned by a direct LU decomposition solver for each element. This numerical approach is very easy to implement, but the computational cost is expensive because we need to consider all the changes of the degrees of freedom in R_c. In order to improve the computational efficiency, we also freeze the element matrices for intervals of time steps. The freezing interval is 5 at the beginning of the computation from the freestream condition, and then linearly increased to a prescribed maximum value $f_{\max}$.

2.3 SV limiter

In order to maintain the numerical stability of the present SV scheme at discontinuities, a data limiting process needs to be implemented. A common approach is to employ a slope

limiter [1, 26] used in the finite volume method. In the development of the SV method for solving the 2D Euler equations [31], a TVB condition was considered to detect a computational cell that is close to a discontinuity. In the detected cell, high-order polynomial reconstructions are replaced by slope limited first-order polynomials that satisfy the TVB constraints.

However, one should note that a slope limiter can yield a distribution of state variables that is considerably different from the original one especially for high-order ($p > 1$) polynomial reconstructions and could degenerate solution accuracy and convergence. In this study, we modify the distribution of state variables near discontinuities according to the weighted sum of the SV averaged value $\bar{P}_i$ and the unlimited reconstruction $\tilde{P}_i$ given by,

$$\hat{P}_i(\vec{r},t) = (1-\Phi_i)\bar{P}_i(t) + \Phi_i\tilde{P}_i(\vec{r},t), \tag{2.18}$$

where $\tilde{P}_i$ is obtained by Eq. (2.2). The limiting coefficient $\Phi_i \in [0,1]$ is determined for both density and pressure profiles so as to satisfy the following relaxed monotonicity constraint,

$$0 < (1-\varepsilon)\bar{P}_i^{\min} \le \hat{P}_i \le (1+\varepsilon)\bar{P}_i^{\max}, \tag{2.19}$$

where $\bar{P}_i^{\min}$ and $\bar{P}_i^{\max}$ are the minimum and the maximum SV cell averaged values among all the neighboring SVs sharing a node with the SV. A small value ε is introduced to retain accuracy near smooth extrema. This condition is virtually the same as that of TVB limiter. The detected discontinuity is, like in the case of TVB scheme, not necessarily be a physical discontinuity, but can be a state jump due to large gradient. In this study, ε is set to be 0.1 for all computations.

The limiting coefficient in Eq. (2.18) can satisfy the constraints (Eq. (2.19)) conveniently if these coefficients are determined by,

$$\begin{aligned}
\Phi_i^+ &= \frac{(1+\varepsilon)\bar{P}_i^{\max} - \bar{P}_i}{\tilde{P}_i^{\max} - \bar{P}_i}, \\
\Phi_i^- &= \frac{(1-\varepsilon)\bar{P}_i^{\min} - \bar{P}_i}{\tilde{P}_i^{\min} - \bar{P}_i}, \\
\Phi_i &= \min\left(\Phi_i^+, \Phi_i^-, 1\right).
\end{aligned} \tag{2.20}$$

In the above expression, the local maximum $\tilde{P}_i^{\max}$ and the minimum $\tilde{P}_i^{\min}$ of the unlimited reconstruction are needed. To search them among all quadrature points is not efficient. We evaluate them from the reconstructed values at 4 vertices of the tetrahedron for the $p1$ approximation and in addition at 6 mid-points of edges for the $p2$ approximation. If Φ_i is smaller than unity, the distribution of state variables in the SV are replaced with the limited one. The CV averaged values are thus altered, although the SV averaged value is unchanged. It is worth noting that a limited reconstruction resulted from Eq. (2.18) still contains a contribution from high order terms in $p2$ approximation unless $\Phi_i = 0$.

2.4 MPI parallelization

The developed SV code has been parallelized with the domain decomposition approach using the Message Passing Interface (MPI) library. At the inter domain boundary, we not only exchange cell averages of conservative variables in adjacent cells but also the max/min data among neighboring cells for density and pressure for the SV limiter. In this study, we restrict the iterative sweeps for the implicit scheme only in each sub-domain. The convergence rate seems not seriously affected by this treatment.

3 Numerical results

In order to demonstrate the fast convergence property of the developed method, several representative test cases are computed. The local Lax-Friedrich flux is applied to obtain the numerical flux for the scalar case, while Roe's approximate Riemann solver [18] is employed for both the Euler and Navier-Stokes cases. The third-order TVD Runge-Kutta scheme [21] with local time stepping is used for explicit time integration to compare the convergence rate with the developed implicit method. All the computations except for the DPW-3 isolated wing case are performed on a Mac Book Pro laptop computer with Intel Core Duo processor (2.0 GHz) and 2 GByte memory.

3.1 Linear wave equation

First, the 3D steady linear advection problem in a cubic domain is considered to examine the spatial accuracy and the convergence property of the present method. The governing equation is given by,

$$\frac{\partial Q}{\partial t} + \frac{\partial Q}{\partial x} + \frac{\partial Q}{\partial y} + \frac{\partial Q}{\partial z} = 0, \quad -1 \leq x, y, z \leq 1. \tag{3.1}$$

A sinusoidal wave pattern given by

$$Q(x,y,z) = \sin \pi (x + y - 2z), \tag{3.2}$$

is imposed at the inflow boundaries $x = -1$, $y = -1$ and $z = -1$. The computed solution is extrapolated at the outflow boundaries $x = 1$, $y = 1$ and $z = 1$. For the initial condition, the variable in the domain is set to be 0. The steady exact solution of this problem is given by Eq. (3.2). In generating computational grids, first an equidistant Cartesian grid of $N \times N \times N$ cells is assumed for the cubic domain and then each cell is further divided into six tetrahedra. In order to assess the spatial accuracy of the present method, three different regular unstructured grids are generated with $N = 10$, 20 and 40.

The L_2 and L_∞ errors are presented in Table 1. It is shown that nearly the formal order of accuracy is achieved for this problem. The convergence rates of the implicit scheme and the explicit one on the medium grid are compared in Fig. 1. Note that we define a

Table 1: Accuracy study of SV schemes for the linear advection problem.

Order of interpolation	Grid	L_2 error	L_2 order	L_∞ error	L_∞ order
p=1	$10\times10\times10\times6$	1.19e-1	–	5.12e-1	–
	$20\times20\times20\times6$	2.72e-2	2.12	1.47e-1	1.80
	$40\times40\times40\times6$	6.23e-3	2.13	3.42e-2	2.10
p=2	$10\times10\times10\times6$	1.56e-2	–	1.41e-1	–
	$20\times20\times20\times6$	2.10e-3	2.90	2.16e-2	2.71
	$40\times40\times40\times6$	3.10e-4	2.76	4.02e-3	2.43

CFL number using the radius of the inscribed sphere of a SV cell as a cell size. In this problem, the CFL number of the implicit scheme was able to be arbitrarily large. The parameters for the implicit scheme are set as $k_{max}=4$ for the number of sweeps, and $f_{max}=10$ for the freezing interval. The speedup factors of the implicit scheme are about 7 in the $p1$ case and 15 in the $p2$ case.

Figure 1: Comparison of the convergence histories between the explicit and the implicit SV schemes for the linear advection problem.

3.2 Inviscid flow over a 3D bump

Inviscid flow over a 3D bump is selected to present the accuracy and the convergence property of the developed code for the Euler equations. The computational domain is defined as,

$$-10\sigma \leq x \leq 10\sigma, \quad f(x) \leq y \leq 10\sigma, \quad -\sigma \leq z \leq \sigma, \tag{3.3}$$

where $f(x)$ is given by the Gaussian distribution function as

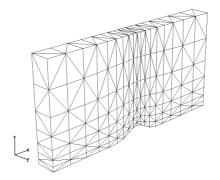$$f(x) = \frac{1}{\sigma\sqrt{(2\pi)}}\exp\left(-\frac{x^2}{2\sigma^2}\right). \tag{3.4}$$

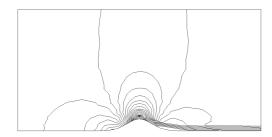Figure 2: Surface grid of the duct with Gaussian distribution (coarse grid, 630 tetrahedra).

We assume the height of the bump as $f(0) = \sigma$, where $\sigma^2 = 1/\sqrt{(2\pi)}$. The computations are carried out on a sequence of grids with 630, 5040 and 40320 cells, by dividing the coarse grid recursively. The bump surface is ensured at refined grid points. The computational domain with a coarse surface grid is shown in Fig. 2. The inflow Mach number of 0.3 is assumed. At the inflow boundary, the total temperature, total pressure, and flow angles ($\alpha = \beta = 0°$) are specified and the outgoing Riemann invariant is evaluated from the flow domain. At the outflow boundary, the static pressure is imposed and the outgoing Riemann invariant, two tangential velocity components, and the entropy are extrapolated. In this problem, Krivodonova's approach [12] is employed for an improved representation of the curved surface. This simplified approach does not consider nonlinear mapping in the reconstruction process and needs only normal vectors of the curved surface to obtain the flux integration. In this study, the bump geometry is represented using piecewise quadratic polynomials for both the $p1$ and $p2$ approximation.

The obtained Mach number contours are shown in Fig. 3. While so-called the entropy layer appears in the $p1$ fine grid solution, almost the symmetric solution is recovered even in the $p2$ coarse grid solution. As reported by Wang et al. [30], if the curved wall is treated as linear facets, residuals do not converge with the coarse grid and a non-symmetric wake structure appears even with the fine grid. This is indeed the case in the present calculation as shown in Figs. 3 (e) and 3 (f).
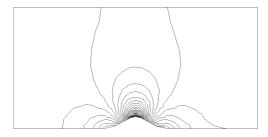
In order to examine the convergence property of the developed method, several different CFL numbers are chosen with the $p1$ or $p2$ approximation on the medium grid. It is found that an impulsive start is not successful for this problem. Therefore, the CFL number is set to 1 at the initial step and increased to the maximum value gradually by the following power law,
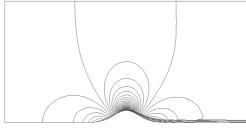
$$CFL^{n+1} = CFL^1 \times a^n, \tag{3.5}$$

where a is set to be 1.2 in this problem. The parameters for the implicit scheme are set as $k_{max} = 3$ and $f_{max} = 20$, respectively. The convergence histories are shown in Fig. 4. For the $p1$ case, a stability limit appears at CFL of about 200. Numerical stability is improved
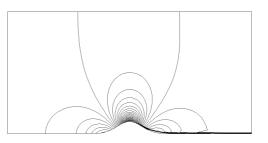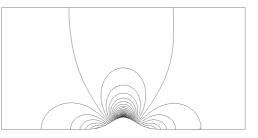
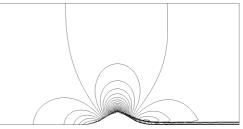(a) $p1$, coarse grid (630 tetrahedra)   (b) $p1$, fine grid (40320 tetrahedra)

(c) $p2$, coarse grid (630 tetrahedra)   (d) $p2$, fine grid (40320 tetrahedra)

(e) $p1$, fine grid (40320 tetrahedra), linear boundary representation   (f) $p2$, fine grid (40320 tetrahedra), linear boundary representation

Figure 3: Mach contours of inviscid flow over a bump.

when the Rusanov's numerical flux [19] is employed for the implicit Jacobian matrices computation. This numerical flux is easily evaluated by replacing the dissipation matrix of the Roe's scheme by the spectral radius of the Jacobian matrix. The convergence history for this case is also plotted in the figure. It is found that an arbitrarily large CFL can be taken when the Rusanov's flux is employed in the evaluation of the Jacobian matrices, though a larger iteration number is required to obtain a converged solution due to the rough assumption of the convective term. The performance is also deteriorated in terms of CPU time. Note that a faster convergence is obtained for smaller CFL when the Jacobian matrices are better approximated using the Roe flux.

For the $p2$ case, a stability limit appears at CFL of about 50 with the Roe's flux. Again, employing Rusanov's flux allows taking arbitrarily large CFL number but a convergence

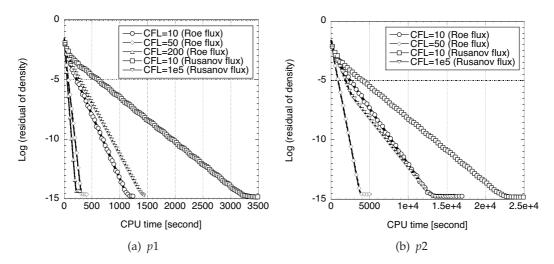(a) *p*1   (b) *p*2

Figure 4: Effect of CFL number on the convergence rate for the inviscid flow over a bump (medium grid).
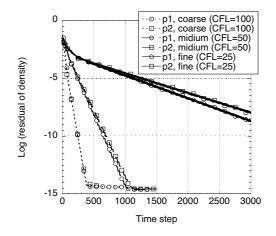

Figure 5: Convergence histories for the inviscid flow over a bump with different grids. The Roe flux is employed to evaluate the implicit Jacobian.

needs more iterations than the case using the Roe's flux with a moderate CFL. The convergence histories for different grids are also compared for the implicit SV schemes in Fig. 5. One can find that a convergence occurs almost independent of the order of solution approximation with the same grid and CFL if the Roe's flux is used in the implicit schemes.

The convergence rate of the present scheme is compared with that of the explicit scheme in Fig. 6. The plotted lines for the implicit scheme are obtained with the maximum CFL of 10^5 using the Rusanov's flux. Assumed CFL numbers for the explicit scheme are 0.5 in the *p*1 and 0.2 in the *p*2 approximation, respectively. The speed up factor is more than an order of magnitude for the *p*1 case. For the *p*2 case, the implicit scheme also shows much faster convergence rate compared to the explicit scheme which needs about two orders of magnitude more CPU time to decrease the residual by two orders of magnitude.
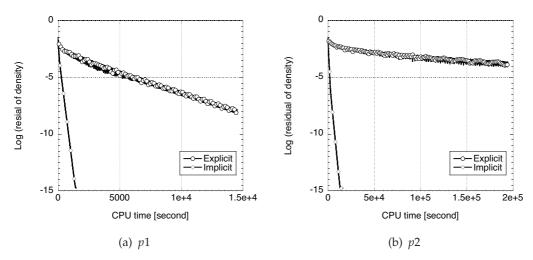
Figure 6: Comparisons of convergence histories between the explicit and implicit schemes for the inviscid flow over a bump (medium grid). The Rusanov flux is employed in the evaluation of the implicit Jacobian with $CFL_{max} = 10^5$.

3.3 Laminar boundary layer flow

The problem of laminar boundary layer flow over a 3D flat plate is selected to verify the present method for solving the Navier-Stokes equations. The convergence rate to steady state is explored. We assume the freestream Mach number of $M = 0.3$ and the Reynolds number based on the plate length of $Re_x = 10,000$. The plate length L is set to be 1.0. At the trailing edge, the thickness of the boundary layer δ estimated by the approximate relation $\delta = 5L/\sqrt{Re_x}$ gives the value of 0.05. The computational domain is then set as follows,

$$-2.5 \leq x \leq 1, \quad 0 \leq y \leq 50\delta, \quad 0 \leq z \leq 0.05. \tag{3.6}$$

The freestream condition is imposed at the inlet boundary ($x = -2.5$) and the static pressure is fixed at the outlet boundary ($x = 1.0$) and also at the top boundary ($y = 50\delta$). A non-slip boundary condition is applied to the wall surface $0 \leq x \leq 1$. A slippery boundary condition is applied to the upstream wall on the bottom boundary and the symmetric boundary condition is enforced at the side wall. A computational grid is generated from a Cartesian grid which has 22 grid points on the viscous wall in the streamwise direction, 8 points in the boundary layer at the trailing edge, and 5 points in the spanwise direction. The surface grid of the generated grid is shown in Fig. 7. Total number of the tetrahedra is 29568.

Computed velocity profiles in the boundary layer with the $p1$ and $p2$ approximation are compared with the Blasius solution in Fig. 8. The obtained profiles agree well with the analytical solution even in the upstream region with less grid resolution, especially for the $p2$ computation.

In implicit computations, the initial CFL number is set to 1 and increased with the
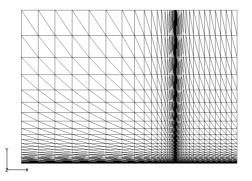
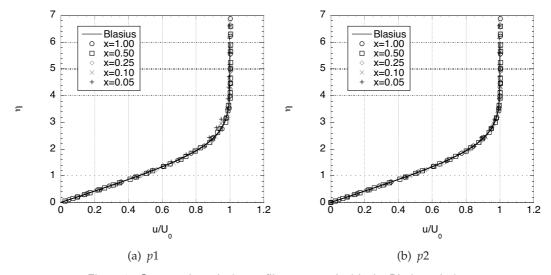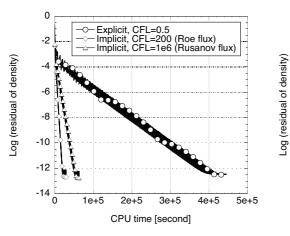Figure 7: Computational grid for the boundary layer flow.

(a) $p1$ (b) $p2$

Figure 8: Computed u velocity profiles compared with the Blasius solution.

power law ($a\!=\!1.05$), with other parameters being $k_{\max}\!=\!3$, $f_{\max}\!=\!40$, and $\delta\!=\!10^{-6}$. The convergence rate of the present method using the $p1$ approximation is compared with the explicit method in Fig. 9. If the Roe flux is used in the implicit Jacobian, the maximum CFL is about 200 for this case. As with the inviscid case in the previous section, employing the Rusanov flux in the Jacobian allows to increase the CFL number to 10^6 or more. The speedup factor of the implicit scheme compared with the explicit scheme (CFL$=$0.5) is more than an order of magnitude. The convergence history of the $p2$ approximation in terms of CPU time is compared with that of the $p1$ approximation in Fig. 10. Almost monotonic convergence to machine zero is also obtained for the $p2$ approximation though with the much longer CPU time. If the Roe flux is used in the implicit Jacobian, the maximum CFL becomes about 100 for this case. Unlike the other cases, however, employing the Rusanov flux in the Jacobian does not allow to increase the CFL number over a thousand.
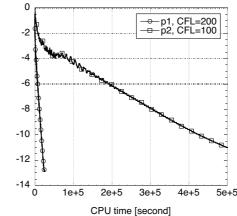
Figure 9: Comparisons of convergence histories for the boundary layer flow using $p1$ approximation.

Figure 10: Comparison of residual histories between the $p1$ and $p2$ approximation for the boundary layer flow.

3.4 High Reynolds number transonic flow over a wing

The present method is applied to obtain high Reynolds number transonic flowfield over an isolated wing by solving the RANS equations with the Spalart-Allmaras turbulence model. The geometry of the wing is taken from the baseline model (wing-1) of the 3rd AIAA Drag Prediction Workshop (DPW-3) [32]. The freestream conditions are $M = 0.76$, $\alpha = 0.5$ [deg], and $Re = 5 \times 10^6$ based on the reference chord length, $c = 197.556$ [mm]. A fully tetrahedral grid is generated with the initial grid spacing from the wall of 0.0052 [mm] ($y^+ \approx 5$), the growth rate of 1.30 and the number of anisotropic cell layer of 25. The outer boundary is located at 100 times the reference chord length away from the origin and the freestream condition is given there. Fig. 11 shows the surface grid on the wing as well as on the root plane. The symmetric flow condition is assumed at the root plane. The generated grid has 2.6 million tetrahedra in the entire domain.

For this case, a preliminary computation is performed with the $p1$ solution approximation. The high order boundary representation is not implemented. In order to capture shock waves without oscillation the SV limiter is employed. Fig. 12 shows the pressure contours plotted on the root plane as well as on the wing surface. One can find that a lambda shaped shock wave pattern develops on the upper surface of the wing. The obtained aerodynamic coefficients are $C_L = 0.479$ and $C_D = 0.0225$. Although the lift coefficient is reasonably well predicted, the drag coefficient is obviously over estimated by more than 10 counts compared with the results in the DPW-3 [32]. This larger drag value can be attributed to the larger off wall grid spacing at the wing surface.

The convergence history for this case is plotted in Fig. 13. In this implicit computation, the CFL number is increased from 1 to 10000 with the power law ($a = 1.01$), with $k_{\max} = 4$ and $f_{\max} = 20$. In order to enhance numerical stability, we employ the Rusanov flux in the implicit Jacobian matrices. The residual reduction of about three and half
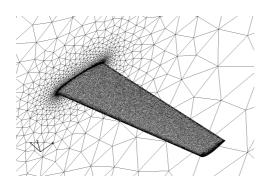
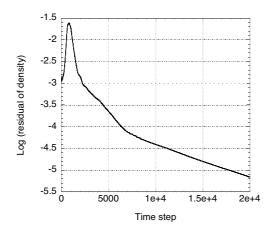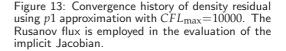Figure 11: Surface grid on the wing-1 of the DPW-3.

Figure 12: Pressure coefficient contours on the wing obtained by $p1$ approximation ($M=0.76$ and $\alpha=0.5$).

Figure 13: Convergence history of density residual using $p1$ approximation with $CFL_{\max}=10000$. The Rusanov flux is employed in the evaluation of the implicit Jacobian.

orders of magnitude is attained after 20000 time steps. The total computational time is about 50 hours using the SGI Altix 32 cores and the required memory was 22 GB, which corresponded to about 1000 words per SV in double precision.

4 Conclusions

A spectral volume method using an implicit LU-SGS solution approach has been developed successfully. First, accuracy study is conducted with linear advection problem and nearly optimal order of accuracy is achieved using $p1$ and $p2$ polynomial reconstructions. The developed method is then applied to the inviscid and the viscous flow problems. For the subsonic inviscid flow over a Gaussian bump, symmetric flow patterns are obtained with the $p2$ solution approximation and the high-order representation of the curved wall surface. For the laminar boundary layer flow over a flat plate, computed velocity profiles using both the $p1$ and $p2$ approximation agree well with the analytical solution. Unlike the linear advection problem, taking a large CFL number (more than one hundred) is not

allowed when the Roe's flux is used in the evaluation of the implicit Jacobian. In order to alleviate this stability limit, we employ more dissipative the Rusanov flux to calculate the implicit Jacobian, and the stability limit of CFL is substantially alleviated. For these test problems, roughly one to two orders of magnitude speedup is observed if compared with the multi-stage Runge-Kutta time integration scheme. Finally, the high Reynolds number transonic flowfield over a wing is computed by solving the RANS equations with the Spalart-Allmaras one-equation turbulence model. The shock wave on the upper wing surface is clearly captured with the developed SV limiter. A fairly good convergence behavior is also indicated. Further improvement in convergence rate on highly clustered viscous meshes probably requires a line implicit solver, which will be examined in our future study.

Acknowledgments

References

[1] T.J. Barth and D.C. Jespersen, The design and application of upwind schemes on unstructured meshes, AIAA Paper, 89-0366 (1989).
[2] F. Bassi and S. Rebay, GMRES discontinuous Galerkin solution of the compressible Navier-Stokes equations, in Lecture Note in Computational Science and Engineering 11, Springer-Verlag, New York, (2000), 197-208.
[3] Q.-Y Chen, Partitions for spectral (finite) volume reconstruction in the tetrahedron, SIAM J. Sci. Comput., 21 (2005), 2352-2380.
[4] R.F. Chen and Z.J. Wang, Fast, block lower-upper symmetric Gauss-Seidel scheme for arbitrary grids, AIAA Journal, 38 (12) (2000), 2238-2245.
[5] B. Cockburn and C.-W. Shu, TVD Runge-Kutta local projection discontinuous Galerkin Finite element method for conservation laws II: general framework, Math. Comput. 52 (1989), 411-435.
[6] B. Cockburn, S. Hou and C.-W. Shu, The Runge-Kutta discontinuous Galerkin method for conservation laws V: Multidimensional systems, J. Comput. Phys. 141 (1998), 199-224.
[7] K. Fidkowski, T. A. Oliver, J. Lu and L. Darmofal, p-Multigrid solution of high-order discontinuous Galerkin discretizations of the compressible Navier-Stokes equations, J. Comput. Phys., 207 (2005), 92-113.
[8] T. Haga, N. Ohnishi, K. Sawada and A. Masunaga, Spectral volume computation of flowfield in aerospace application using earth simulator, AIAA paper, 2006-2823, (2006).

  [9] R. Harris, Z.J., Wang and Y. Liu, Efficient quadrature-free high-order spectral volume method on unstructured grids: Theory and 2D implementation, J. Comput. Phys., 227 (2008), 1620-1642.
[10] A. Jameson and S. Yoon, Lower-upper implicit schemes with multiple grids for the Euler equations, AIAA Journal, 25 (7) (1987), 929-935.
[11] R. Kannan, Y. Sun, and Z.J. Wang, A study of viscous flux formulations for an implicit p-multigrid spectral volume Navier Stokes solver, AIAA paper, 2008-783, (2008).
[12] L. Krivodonova and M. Berger, High-order accurate implementation of solid wall boundary conditions in curved geometries, J. Comput. Phys., 211 (2006), 492-512.
[13] Y. Liu and M. Vinokur, Exact integration of polynomials and symmetric quadrature formulas over arbitrary polyhedral grids, J. Comput. Phys., 140 (1998), 122-147.
[14] Y. Liu, M. Vinokur and Z.J. Wang, Spectral difference method for unstructured grids I: Basic formulation, J. Comput. Phys., 216 (2006), 780-801.
[15] Y. Liu, M. Vinokur and Z.J. Wang, Spectral (finite) volume method for conservation laws on unstructured grids V: Extension to three-dimensional systems, J. Comput. Phys., 212 (2006), 454-472.
[16] H. Luo, J.D. Baum and R. Löhner, A *p*-multigrid discontinuous Galerkin method for the Euler equations on unstructured grids, J. Comput. Phys., 211 (2006), 767-783.
[17] P. Rasetarinera and M. Y. Hussaini, An efficient implicit discontinuous spectral Galerkin method, J. Comput. Phys., 172 (2001), 718-738.
[18] P.L. Roe, Approximate Riemann solvers, parameter vectors, and difference schemes, J. Comput. Phys., 43 (1981), 357-372.
[19] V.V. Rusanov, Calculation of interaction of non-steady shock waves with obstacles, J. Comput. Math. Phys., 1 (1961), 267-279.
[20] D. Sharov and K. Nakahashi, Reordering of hybrid unstructured grids for lower-upper symmetric Gauss-Seidel computations, AIAA Journal, 36 (3) (1998), 484-486.
[21] C. -W. Shu, Total-variation-diminishing time discretizations, SIAM J. Sci. Stat. Comput., 9 (1988), 1073-1084.
[22] Y. Sun and Z.J. Wang, Formulations and analysis of the spectral volume method for the diffusion equation, Commun. Numer. Meth. Engng., 20 (2004), 927-937.
[23] Y. Sun, Z.J. Wang and Y. Liu, Spectral (finite) volume method for conservation laws on unstructured grids VI: Extension to viscous flow, J. Comput. Phys., 215(1) (2006), 41-58.
[24] Y. Sun, Z.J. Wang and Y. Liu, Efficient implicit non-linear LU-SGS approach for compressible flow computation using high-order spectral difference method, Commun. Comput. Phys., 5 (2009), 760-778.
[25] K. Van den Abeele and C. Lacor, An accuracy and stability study of the 2D spectral volume method, J. Comput. Phys., 226 (2007), 1007-1026.
[26] V. Venkatakrishnan, Convergence to steady state solutions of the Euler equations on unstructured grids with limiters, J. Comput. Phys., 118 (1995), 120-130.
[27] Z.J. Wang, Spectral (finite) volume method for conservation laws on unstructured grids: basic formulation, J. Comput. Phys., 178 (2002), 210-251.
[28] Z.J. Wang and Y. Liu, Spectral (finite) volume method for conservation laws on unstructured grids II: extension to two-dimensional scalar equation, J. Comput. Phys., 179 (2002), 665-697.
[29] Z.J. Wang and Y. Liu, Spectral (finite) volume method for conservation laws on unstructured grids III: one-dimensional systems and partition optimization, J. Sci. Comput., 20 (2004), 137-157.
[30] Z.J. Wang and Y. Liu, Extension of the spectral volume method to high-order boundary

representation, J. Comput. Phys., 211 (2006), 154-178.

[31] Z.J. Wang, L. Zhang and Y. Liu, Spectral (finite) volume method for conservation laws on un-structured grids IV: extension to two-dimensional systems, J. Comput. Phys., 194(2) (2004), 716-741.

[32] http://aaac.larc.nasa.gov/tsab/cfdlarc/aiaa-dpw/